WHAT IS CLAIMED IS:

1. A system for indicating status of a storage device, comprising:

N storage locations of a memory device, having *T* tag bits;

N-bit read and write registers in communication with the corresponding *N* storage locations; and

logic for comparing the contents of the read and write registers with corresponding contents of the storage locations to determine which of the *N* storage locations have been written to in lieu of being read from, and the status of such *N* storage locations dependent on whether the respective *T* tag bits are active.

2. The system as recited in claim 1, further comprising first and second clocks, such that data are written to the *N* storage locations synchronously with the first clock, and read from the *N* storage locations synchronously with the second clock.

3. The system as recited in claim 2, further comprising combinatorial logic that compares the respective contents of the write and read registers to determine which of the *N* storage locations have been written to but not read.

4. The system as recited in claim 3, wherein the combinatorial logic comprises an exclusive OR (XOR) gate associated with each of the *N* storage locations, and such that the i^{th} XOR gate compares the i^{th} bit in the write register with the i^{th} bit in the read register.

5. The system as recited in claim 3, wherein "valid" storage locations comprise those that have been written to, but not read.

6. The system as recited in claim 5, further comprising combinatorial logic configured to transmit the logic level of each of the T tag bits in each valid storage location.

7. The system as recited in claim 6, wherein the combinatorial logic comprises a set of T AND gates for each of the N storage locations, and such that the i^{th} gate in the set associated with the j^{th} storage location ANDs the i^{th} tag bit in the j^{th} storage location with the output of the j^{th} XOR gate.

8. The system as recited in claim 7, further comprising T combinatorial gates, wherein the i^{th} gate ORs together the output of the i^{th} AND gate in each of the N sets of T AND gates.

9. The system as recited in claim 8, further comprising circuitry for synchronizing the ORed tag bit logic levels with the second clock.

10. A system for indicating the status of a first-in first-out (FIFO) data buffer, wherein the FIFO comprises:

N data locations, each location adapted to receive D data bits and T tag bits;

first and second clocks, such that data are written to the FIFO locations synchronously with the first clock and read from the FIFO synchronously with the second clock; and

and wherein the system detects the status of tag bits in any FIFO locations to which data have been written to in lieu of being read from.

11. The system as recited in claim 10, further comprising N-bit read and write registers, wherein the write register records which FIFO locations have been written to, and the read register records which locations have been read.

12. The system as recited in claim 11, further comprising combinatorial logic that compares the respective contents of the read and write registers to determine which FIFO locations have been written to but not read.

13. The system as recited in claim 12, wherein "valid" locations comprise those that have been written to, but not read.

14. The system as recited in claim 13, further comprising combinatorial logic configured to transmit the logic level of the T tag bits in each FIFO location that is valid, and to suppress the logic level of the tag bits in each FIFO location that is not valid.

15. The system as recited in claim 14, further comprising combinatorial logic configured to generate T outputs, such that the i^{th} output represents the logical OR of the i^{th} tag bit from each of the valid FIFO locations.

16. A method for indicating status of a storage device, wherein the storage device comprises:

N storage locations, each location having T tag bits; and

first and second clocks, such that data are written to the storage locations synchronously with the first clock and data are read from the storage locations synchronously with the second clock;

and wherein the method comprises:

detecting storage locations to which data have been written, but not read (i.e., valid storage locations);

detecting active tag bits within valid storage locations; and

generating a logic signal for each of the T tag bits, such that the i^{th} logic signal is active if the i^{th} tag bit is active in any of the valid storage locations.

17. The indicator method as recited in claim 16, wherein detecting valid storage locations comprises:

associating with each storage location a flag in a first register and a flag in a second register;

toggling the associated flag in the first register when data are written to a given storage location;

toggling the associated flag in the second register when data are read from a given storage location; and

comparing the state of the associated flag in the first and second registers to determine whether a particular storage location has been written to but not read.

18. The indicator method as recited in claim 16, wherein detecting active tag bits within valid storage locations comprises, for each storage location:

comparing the associated first and second register flags to determine whether the storage locations is valid; and

ANDing the result of the flag comparison with the logic state of each tag bit in the storage location.

19. The indicator method as recited in claim 16, wherein generating a logic signal for the i^{th} tag bit comprises ORing the results of the active tag detection for the i^{th} tag bit from all of the storage locations.

20. The indicator method as recited in claim 16, further comprising synchronizing the logic signal generated for each tag bit to the second clock.